

EXPRESS MAIL LABEL NO: EL707907814US

METHOD AND APPARATUS FOR COMPONENT PLAN ANALYSIS
UNDER UNCERTAINTY

Pascal Bensoussan,

Paul Dagum,

Adam Galper,

Michael Goldbach,

Balazs Kralik

and

Vivek Vaidya

Technical Field of the Invention

The present invention is generally related to resource-planning methods used by manufacturing companies and other organizations. More specifically, the present invention includes a method for strategic resource planning that accounts for uncertainties inherent in the forecasting process.

Background of the Invention

Strategic resource planning is an essential tool for many companies and organizations. Strategic resource planning allows companies to acquire the resources that they need in the amounts that they need, when they are needed. These resources can include goods, raw materials, services, manpower, facilities and a wide range of other things.

The value of strategic resource planning depends entirely on its accuracy. Inaccurate forecasting is, in many cases, worse than no forecasting at all. As a result, a great deal of energy has been invested in creating accurate methods for forecasting. Unfortunately, forecasts will

always be imperfect. Therefore, the planner will always be faced with the financial risks associated with over- and under-supply. The real business challenge is to effectively manage the risks of planning under uncertainty.

5 Risk management requires explicit consideration of multiple possible outcomes and their likelihood (or probability) of occurring. In this case, multiple possible outcomes arise because the demand forecast is not clairvoyant. The probability that any particular outcome will occur is related to the uncertainty around the demand forecast. Existing systems do not treat the uncertainty of demand forecasts explicitly. Instead of explicitly quantifying these uncertainties, existing systems merely assume that demand forecasts may be inaccurate. As a result, the degree
10 of uncertainty is never elicited for use in a formal analysis. When the underlying demand data is not represented probabilistically, risk management methods are difficult if not impossible.

15 Resource planning for resources consumed in the assembly (creation or manufacture) or refinements (from which value is derived) is a particular complex task as demand for resources is actually driven by demand for the refinements. Moreover, while costs are associated with positioning resources, it is ultimately the availability of refinements that provides revenue.

20 Demands for finished goods often have relationships that can be described as neutral, cannibalistic or synergistic. Goods that have synergistic relations tend to boost each other's sales. This happens when the products tend to be used together, such as a computer system and monitor. Goods that have cannibalistic relations compete for sales. This happens when one good is used in place of the other, such as a computer system and a slightly faster computer system. Unfortunately, existing systems and methods for strategic resource planning do not capture explicitly or use the demand relations among finished goods. As a result, these systems tend to produce sub-optimal results.

25 Existing systems do not explicitly capture time based resource availability in the form of contractual terms that provide supply flexibility around a specific resource plan. Very often planned quantities of resources for refinement can be canceled (at costs), or additional quantities can be expedited (at costs) as a reaction to real demand.

Existing systems calculate resource demands from refinement demands and may provide a measure of refinement availability from resource availability (given resource allocation rules).

However, existing systems assume demand information to be deterministic (single forecast estimate). Existing systems are not able to connect demands for refinements and implied demands for resources, as well as availability of resources and implied availability of refinements.

- 5 Finally, existing systems do not have the ability to provide performance measures on cost exposure, revenue, profit and availability risks at both refinement and resource levels that account for the uncertainty and demand dynamics for refinements competing for scarce resources.

Summary of the Invention

10 An embodiment of the present invention includes a method and apparatus for component plan analysis under uncertainty. During the first step of this method, assumptions about products and components are captured in an entity called a "scenario." A scenario is the parameterization of all the demand, financial, and operational information for a portfolio of products and components across a set of time buckets (planning periods).

15 The second step of the method is to specify the component plan that will be analyzed. The component plan identifies the quantities of each resource that will be positioned during each planning period.

20 The third step of the method is to generate a request for analysis. The request for analysis specifies parameters that will be used in order to evaluate the performance and the risks associated with a selected component plan (see step two) under a specific scenario (see step one). The parameters specify such things as, for example, whether the component plan is a commitment to use or procure the resources, or whether the acquisition of resources can be responsive to demand fluctuations within flexibility ranges defined by contractual terms.

25 The fourth step of the method is to calculate risk indicators. During this step, an analytical engine processes the request for analysis. The analytical engine calculates all the performance indicators and returns the results. These results are then stored in a database or other persistent storage system.

The fifth step of the method is to display the results of the calculations to the user.

Brief Description of the Drawings

For a more complete understanding of the present invention and for further features and advantages, reference is now made to the following description taken in conjunction with the accompanying drawings, in which:

5 Figure 1 is a block diagram of an Internet-like network shown as a representative environment for deployment of the present invention.

Figure 2 is a block diagram of a computer system as used within the network of Figure 1.

Figure 3 is a block diagram of an analysis system as used by an embodiment of the present invention.

10 Figure 4 is a flowchart showing the steps associated with an embodiment of the risk analysis method of the present invention.

Figure 5 is a flowchart showing the sub-steps associated with the assumption specification step of Figure 4.

15 Figure 6 is a flowchart showing the sub-steps associated with the component plan specification step of Figure 4.

Figure 7 is a block diagram showing a representative set of database tables used to store the plan and scenario generated during the method shown in Figure 4.

Figure 8 is a flowchart showing the sub-steps associated with the analysis request generation step of Figure 4.

20 Figure 9 is a flowchart showing the sub-steps associated with the step of calculating and storing risk factors of Figure 4.

Figure 10 is a flowchart showing the sub-steps associated with the analytic engine invocation step of Figure 9.

25 Figure 11 is a flowchart showing the sub-steps associated with the step of computing performance and risk indicators of Figure 9.

Figure 12 is a flowchart showing the sub-steps associated with the step of returning and storing performance and risk indicators of Figure 9.

Figure 13 is a block diagram showing a representative set of database tables used to store the storing performance and risk indicators generated during the method shown in Figure 4.

5 Figure 14 is a block diagram showing use of the risk analysis environment provided by an embodiment of the present invention.

Detailed Description of the Preferred Embodiments

10 The preferred embodiments of the present invention and their advantages are best understood by referring to Figures 1 through 4 of the drawings. Like numerals are used for like and corresponding parts of the various drawings.

Definitions

15 Component: in this document, the resources required to produce products are referred to as components. It should be noted that components may be sold directly as products in some cases.

Product: goods or other refinements of components are described as products.

20 Position: positioning is an alternative to ordering components. When a company (of other entity) positions a component, it has arranged for it to be available without actually putting the component in inventory. Thus, as an example, a supplier may agree to provide a certain quantity of a component during a time frame. The component is positioned in that quantity for that time frame.

Component plan: a list of quantities for each component, representing a company's component position for a given planning period.

25 Scenario: a set of assumptions about products and components. A scenario includes at least product parameters, component parameters, and component consumptions.

Environment

In Figure 1, a computer network 100 is shown as a representative environment for an embodiment of the present invention. Computer network 100 is intended to be representative of the complete spectrum of computer network types including Internet and Internet-like networks.

Computer network 100 includes a number of computer systems, of which computer system 102a through 102f are representative. Computer systems 102 are intended to be representative of the wide range of large and small computer and computer-like devices that are used in computer networks of all types. Computer systems 102 are specifically intended to include non-traditional computing devices such as personal digital assistants and web-enabled cellular telephones.

Figure 2 shows a representative implementation for computer systems 102. Structurally, each computer system 102 includes a processor, or processors 200, and a memory 202. Processor 200 can be selected from a wide range of commercially available or custom types. An input device 204 and an output device 206 are connected to processor 200 and memory 202. Input device 204 and output device 206 represent all types of I/O devices such as disk drives, keyboards, modems, network adapters, printers and displays. Each computer system 102 may also include a disk drive 210 of any suitable disk drive type (equivalently, disk drive 210 may be any non-volatile mass storage system such as "flash" memory).

Software Architecture

For the purposes of description, it may be assumed that computer network 100 includes an analysis system hosted by one or more computer systems 102. A representative implementation for a system of this type is shown as analysis system 300 of Figure 3. Analysis system 300 includes an analytic application server ("server") 302 that coordinates multiple, distributed analytic engines ("engines") 304.

Server 302 interacts with a database 306 and any number of web browsers 308. Server 302 manages web interactions, database access, XML-based data exchange, report design and delivery, and asynchronous messaging among engines. The server can be based, for example, on Java 2 Enterprise Edition (J2EE) standards.

Analytic engines 304 are stateless components offering a set of related analytics. Clients submit problem statements to analytic engines 304. For the particular implementation being

described, clients submit problem statements as XML documents. For other implementations, other encodings and protocols may be used. Each problem statement identifies an analysis to perform and includes all necessary input data. Analytic engines 304 output a solution as XML documents. Once again, the use of XML is implementation dependent—other implementations
 5 may use other encodings and protocols. Each analytic engine 304 must define the structure/schema of its expected input and output.

Each engine 304 includes an engine interface and an engine core. The engine interface handles XML-based I/O, message queue management, and provides standards-based APIs. The engine core processes the analytic requests for a set of related problems. For the particular
 10 implementation being described, engine interfaces are implemented as Java applications and engine cores implemented as compiled libraries of Matlab source code. Each engine interface can communicate with its corresponding engine core through, for example, Java Native Interface (JNI) calls in which input and output filenames are passed.

Overview of Method and Apparatus for Component plan Analysis Under Uncertainty

15 An embodiment of the present invention includes a method and apparatus for component plan analysis under uncertainty. A representative implementation of this embodiment is shown in Figure 4 as Method 400. In this particular implementation of this embodiment, the term “product” will be used for “refinement” and the term “component” will be used for “resource”. Method 400 begins with step 402 where a planner captures assumptions about products and
 20 components in an entity called a “scenario.” A scenario is the parameterization of all the demand, financial, and operational information for a portfolio of products and components across a set of time “buckets” (planning periods). The data input process can be either manual (through a user interface) or automated (e.g., by importing data from external systems).

After completing step 402, Method 400 continues with step 404. In step 404, the planner
 25 specifies a component plan to be analyzed. The component plan identifies the quantities of each resource that will be used or procured during each planning period.

In step 406, the planner generates a request for analysis. The request for analysis specifies the parameters that will be used in order to evaluate the performance and the risks associated with the component plan of step 404 and the scenario of step 402.

In step 408, the request for analysis is submitted to an analysis engine for calculation of risk indicators. The analytical engine calculates all the performance indicators and returns the results. The results are then typically stored in a database or other persistent storage system.

Assumption Specification

5 The process of assumption specification (step 402 of Figure 4) can be further subdivided into the series of steps shown in Figure 5. In the first of these steps, step 402-2, the user of the system specifies assumptions about product parameters. These parameters describe the products that are to be produced and can be manually input or loaded from external systems. Typical product parameters include demand parameters, financial parameters, and demand support parameters.

10 Demand parameters characterize the demand forecast for products on a period-by-period basis. This demand is characterized in a probabilistic sense. In the preferred embodiment of this invention, demand is characterized by the mean forecast and standard deviation of the demand for each product for each period.

15 Financial parameters characterize the financial aspects of each product. This is done on a period-by-period basis and typically includes the revenue/income that is gained from satisfying the product demand (i.e. product selling price) and the cost of refining the product (i.e. including the standard costs of consumed components).

20 Demand support parameters characterize product availability targets, on a period-by-period basis. Under assemble-to-order (ATO) production with shared components, production is probabilistic because the actual number of units produced will depend on demand for all products that share components. Therefore demand support parameters are typically expressed in terms of:

25 Product target support level – This parameter defines the number of units of a product that can be produced at a specified level of confidence (e.g., 800 widgets may be produced with a confidence of 85%); and

 Product target level support confidence – This parameter defines the probability that a given plan can meet a given product target support level (e.g., there is a confidence of 85% that 800 widgets may be produced).

In step 402-4 of Figure 5 the planner specifies product interactions. Product interactions describe the relationship (neutral, cannibalistic or synergistic) between different products. Thus, if sales of one product encourage sales of a second, a positive correlation coefficient is used to describe the relationship between the first and second products. Zero or negative correlation coefficients are used to describe neutral or cannibalistic relationships between products. Product interactions are described on a period-by-period basis.

In step 402-6 of Figure 5 the planner specifies component parameters. Component parameters characterize the components that are required for the products that are to be produced. These parameters typically include component supply parameters and component financial parameters. Component supply parameters characterize the planned replenishment of components on a period-by-period basis. Generally, this characterization includes the component availability constraints for each required component (e.g., the minimum and maximum supply quantities for each component). Component supply parameters also typically specify the lead-time for each component on a period-by-period basis. The lead-time for a component is the time delay between ordering the component and having the component available for manufacturing or resale.

Component supply parameters may also include parameters that describe the contractual basis under which each component is obtained. These parameters are typically expressed in terms of:

cancellation maximums – The maximum quantity of a component plan that can be returned to a supplier (with or without cancellation fees).

cancellation threshold – The maximum quantity of a component plan that can be returned to a supplier for a full refund.

expedite threshold – The maximum quantity of a component that a supplier will provide, beyond the quantity initially planned, without imposing expediting fees.

expedite maximum – The maximum quantity of a component that can be obtained (with or without expediting fees).

The component financial parameters specified by the planner characterize the costs associated with each component. Typically, these parameters include: the procurement cost of the component (i.e. standard cost), the cost of canceling part of the component plan below the cancellation threshold and the cost of procuring additional components above the expedite threshold.

In step 402-8 of Figure 5 the planner specifies component consumption often associated with the product bill-of-materials (BOM). In this step, the consumption relationship between each product and all of its consumed components is specified on a period-by-period basis. This describes, in particular, how many of each component is required to produce each product.

In step 402-10 of Figure 5 the information generated in steps 402-2 through 402-8 is stored in a persistent way for later use. Typically, this is done by storing the information in database tables. Other forms of persistent storage can also be used.

Component plan Specification

The process of component plan specification (step 404 of Figure 4) can be further subdivided into the series of steps shown in Figure 6.

In step 404-2 of Figure 6 the planner creates a component plan. The component plan contains the intended procurement quantities for each component for each planning period. Component plan quantities are measured in the standard units of measure for the components. For example, electronic components are measured in units, and assembly capacity is measured in hours. The component plan can be created in a number of different ways:

One method is to create the component plan using a default plan loaded from the database or other persistent storage (either from the database 306 or from external systems).

Another method is to copy an existing component plan. This method is particularly useful where a similar plan is already in existence, such as is the case where a manufacturing process is being revised or optimized or if the user would like to compare the performance of a component plan with a modified copy of the plan under the same scenario.

Still another method is to generate an optimal component plan from a specified scenario (such as the scenario defined in step 402). For this type of plan creation method, the specified scenario is evaluated to create component plans that satisfy certain objectives, e.g. maximize total profit or total revenue or minimize total costs (with or without budget, component supply, or product service level constraints).

In step 404-4 of Figure 6 the planner updates the levels included in the component plan. This can be done manually from a user interface. In other cases, updated levels are obtained automatically. Automatic updates can be obtained from external systems (such as ERP, SCM, or Planning Systems) or by optimizing scenarios for profit, cost, budget or revenue. Automatic updates can also be obtained by overwriting with plan levels from another plan. Enterprise Resources Planning (ERP) systems are used to capture data and manage data entry to support both the manufacturing and accounting operations. Supply Chain Management (SCM) applications are used to manage information across supply chains around product data management (PDM), supplier relationship management, master production schedules (MPS), and replenishments.

In step 404-6 of Figure 6 the information generated in steps 402-2 through 402-8 is stored in a persistent way for later use. Typically, this is done by storing the information in database tables. Other forms of persistent storage can also be used.

Figure 7 shows a representative set of database tables that may be used for this purpose.

Table 702 contains information about a scenario document (name, description, date of creation, date of last modification, and other parameters that may be used by the system).

Table 704 contains all product parameters listed above for a given product under a given scenario in a given planning period.

Table 706 contains all component parameters listed above for a given component under a given scenario in a given planning period.

Table 708 contains all consumption information between a product and a given component (typically called a Bill of Materials, or BOM, in manufacturing environments), under a given scenario in a given planning period.

Table 710 contains product demand interactions for two products under a given scenario in a given planning period.

Table 712 contains information about a component plan document (name, description, date of creation, date of last modification, and other parameters that may be used by the system).

5 Table 714 contains plan information (quantity, and other parameters that can be used by the system) for a given component, in a given plan document in a given planning period.

Analysis Generation

The process of generating an analysis (step 406 of Figure 4) can be further subdivided into the series of steps shown in Figure 8. In the first of these steps, step 406-2, the planner fills out a request for analysis form. The planner creates a request for analysis form in order to evaluate the performance and the risks associated with a selected plan under a specific scenario. For the particular implementation being described, the request for analysis form includes a combination of the following fields:

15 Name: The name provides a method for referring to the analysis. For most implementations, this field is required.

Description: This field allows the planner to enter additional text or notes to accompany the analysis. This field is typically optional.

Scenario: The planner is required to select a scenario from a list of existing scenarios.

Plan: The planner is required to select a plan from a list of existing plans.

20 Mode: This field allows the planner to choose between Commit and Respond modes of analysis, where Commit and Respond modes are defined as follows:

Commit means that the analysis is to be performed using the assumption that planner is committed to purchasing the components as planned and cannot take advantage of expediting or cancellation options in the future when actual demand will be known. In commit mode, the analysis will not include these component parameters in its calculations. This is the default selection.

Respond means that the analysis is to be performed using the assumption that expediting or cancellation options can be used.

Confidence Level %: This field allows the planner to choose between using a default confidence level and specifying a global confidence level as follows:

5 To enter a global confidence level, the planner enters a numerical percentage in the confidence level field. The analysis then applies this value for all products.

To use a default confidence level, the planner selects (or doesn't modify) the default selection in the confidence level field. The analysis then applies the confidence levels specified in the product parameters of the selected scenario.

10 Status Notification: The planner uses this field to identify an address to use for email notification. Email is sent to this address when the risk analysis is complete.

In step 406-4 of Figure 8 the planner submits the request for analysis generated in the previous step. Typically, this is done by pressing a submit button or other user interface icon. In some cases, default value may be supplied for the required inputs of step 406-2. This allows the user to skip step 406-2 and proceed directly to step 406-4. This provides a single step (one-click) method for default values.

Calculation of Risk Indicators

In step 408, the request for analysis is submitted to an analysis engine for calculation of risk indicators. This process can be further subdivided into the series of steps shown in Figure 9.

20 In the first of these steps, step 408-2, server 302 invokes one of analytic engines 304. In step 408-4, the selected analytic engine 304 performs a risk analysis with the specified parameters. In step 408-6, the results of this analysis are returned to server 302 and stored in database 306.

Step 408-2 can be further subdivided into the series of steps shown in Figure 10. In the first of these steps, step 408-2-2, the client requests that the analysis be performed. As discussed, 25 this request takes the form of a specified plan, specified scenario and specified analysis parameters.

In step 408-2-4, server 302 queries database 306 to retrieve the specified plan and specified scenario. In step 408-2-6, server 302 encodes the retrieved plan and scenario, along with the specified analysis parameters as an XML message. Server 302 sends the XML message to analytic engines 304 using, for example, a JMS (Java Message Service) queue. In step 408-2-8, the XML message is shown as XML input for one of analytic engines 304. A representative example of a DTD (Document Type Definition) specifying the format of an XML message of this type is attached to this document as Appendix A.

Step 408-4 can be further subdivided into the series of steps shown in Figure 11. In the first of these steps, step 408-4-2, one of analytic engines 304 queries the JMS queue where the XML message is queued. In general, analytic engines 304 query the JMS queue when they become idle and are available to perform work. Step 408-4-2 corresponds to the case where one of analytic engines 304 has become idle and is performing this query. The available analytic engine 304 retrieves the XML message generated by server 302 as part of this query.

Each analytical engine 304 is subdivided into an engine interface and an engine core. The engine core is designed to accept analysis requests from the engine interface. The engine interface is designed to map a particular language (in this case XML) to input understood by the engine core. This provides clean separation between the engine core and the server 302 allowing flexibility to define multiple interfaces to the engine core so that analytical engines 304 can with a range of different input languages.

In step 408-4-4, the engine interface of the selected analytic engine 304 validates the XML message generated by server 302 to ensure that the XML message does not include errors. The engine interface then translates the XML message generated by server 302 into a text input for the engine core.

In step 408-4-6, the engine interface of the selected analytic engine 304 sends the translated XML message to its engine core. For the described implementation, this is done using a JNI call. For other implementations, other message technologies may be used.

In step 408-4-8, the translated XML message is shown as a text input for the engine core of the selected analytical engine 304.

In step 408-4-10, the engine core processes the analysis request included in the text input. The engine core performs the calculations of the performance and risk indicators. Any analytical method could be used to generate the risk metrics. For example, methods for computing expected erosion, cancellation and expediting fees are disclosed in a co-pending US Provisional Application Serial No.: 60/229,611 entitled "Method and Business Process for Estimation of Erosion Costs in Assemble-to-Order Manufacturing Operations." A method for computing component gating risk is disclosed in a co-pending US Provisional Application Serial No.: 60/229,840 entitled "Method and Business Process for Estimation of Component Gating and Shortage Risks in Assemble-to-Order Manufacturing Operations." The disclosure of both of these documents is incorporated herein by reference.

The engine core produces its results as text output. In step 408-4-12, the engine core sends its output to the engine interface. For the particular implementation being described, the engine output is text. It should be appreciated, however, that different encodings and protocols could be used to perform this step. For the described implementation, the engine performs the output step by making a JNI call. For other implementations, other messaging technologies may be used.

In step 408-4-14, the text output is shown as input for the engine interface of the selected analytical engine 304.

In step 408-4-16, the engine interface of the selected analytic engine 304 translates the text output of the engine core into a second XML message. The second XML message encodes the results computed by the engine core. In step 408-4-18, the engine interface sends the second XML message back to server 302. The engine interface sends the second XML message to server 302 using the JMS queue. In step 408-2-8, the second XML message is shown as XML input for server 302. A representative example of a DTD specifying the format of an XML message of this type is attached to this document as Appendix B.

Step 408-6 can be further subdivided into the series of steps shown in Figure 12. In the first of these steps, step 408-6-2, server 302 queries the JMS queue where the second XML message is queued. Server 302 retrieves the second XML message as part of this query.

In step 408-6-4, server 302 stores the results of the computations performed by the selected analytic engine 304 in database 306. Typically, this information is stored as database tables. Other forms of persistent storage can also be used. Figure 13 shows a representative set of database tables that may be used for this purpose. The analytic results are saved in database 306
 5 for subsequent reporting and navigation.

Table 1302 contains all the information about the analysis (folder where it is placed, status indicating if it was completed successfully or if it failed, process time for the computation, and other parameters that may be used by the system).

Table 1304 contains all the information about the folder used to support the resource
 10 planning process (name, creation date, beginning and ending planning periods, and other parameters that may be used by the system).

Table 1306 contains all the information about the scenario (same as 702).

Table 1308 contains all the information about the component plan (same as 712).

Table 1310 contains all the information referencing the input parameters to the analysis
 15 (scenario specified in step 402, component plan specified in step 404, and analysis parameters specified in step 406).

Table 1312 contains all the risk and performance indicators calculated by the analysis for each component and for each planning period.

Table 1314 contains all the risk and performance indicators calculated by the analysis for
 20 each product and for each planning period.

Error Handling

There are several kinds of errors that engine 304 may return:

Parsing errors: The input data is improperly formatted,

Validation errors: The input data does not pass data validation checks (i.e. data out of
 25 range or inconsistent),

Analytic errors: Engine 394 experienced an internal failure.

All errors are packaged as XML documents for return to server 302.

Risk Analysis Navigation

In many cases, it is desirable to include a risk analysis environment as part of the present invention. The risk analysis environment is a user interface for navigating the performance and risk indicators stored in database 306. This allows the planner to navigate and analyze performance and risk indicators and to make modifications to the corresponding plans and scenarios. The risk analysis environment is preferably implemented to address the following three requirements:

Ability to provide both product and component risk summary views and product-to-components and component-to-products detailed views,

Ability to navigate logically between views with the simple click of a mouse,

Ability for users to configure their own sets of views.

To address these requirements, the risk analysis environment includes a series of data presentation screens. For typical implementations, these include a product summary screen, a component summary screen, a component detail screen, and a product detail screen. Use of these screens is shown in Figure 14. As shown, use of the risk analysis environment typically begins with the planner selecting a particular risk analysis to analyze. Once that selection is complete, the planner can select to continue the analysis at either the component or product level. Each level allows the user to select a particular component or product (respectively). The planner can then continue at the summary or detail level for the selected component or product. The process can be repeated for different products, components or risk analyses.

Planners use the risk analysis environment to adjust plans and scenarios to specific objectives such as product support and revenue targets, inventory cost containment, and revenue loss risk minimization. The risk analysis method of Figure 4 through 13 can then be re-executed and the results analyzed within the risk analysis environment.

The risk analysis environment is customizable, allowing users to select aspects (input and output parameters) of components and products and include these aspects in user-defined views. These user-defined views would then be linked to extend the navigation example of Figure 14.

5 Although particular embodiments of the present invention have been shown and described, it will be obvious to those skilled in the art that changes and modifications may be made without departing from the present invention in its broader aspects, and therefore, the appended claims are to encompass within their scope all such changes and modifications that fall within the true scope of the present invention.